NASA Technical Memorandum 104003

Scene Segmentation of Natural Images Using Texture Measures and Back-Propagation

Banavar Sridhar, Anil Phatak, and Gano Chatterji

March 1993

Quick Release – This Technical Memorandum is an unedited report. It is being released in this format to quickly provide the research community with important information.

National Aeronautics and Space Administration

NASA Technical Memorandum 104003

Scene Segmentation of Natural Images Using Texture Measures and Back-Propagation

Banavar Sridhar, Anil Phatak, and Gano Chatterji
Ames Research Center, Moffett Field, California

March 1993

National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, California 94035-1000

SUMMARY

Knowledge of the three-dimensional world is essential for many guidance and navigation applications. A sequence of images from an electro-optical sensor can be processed using optical flow algorithms to provide a sparse set of ranges as a function of azimuth and elevation. A natural way to enhance the range map is by interpolation. However, this should be undertaken with care since interpolation assumes continuity of range. The range is continuous in certain parts of the image and can jump at object boundaries. In such situations, the ability to detect homogeneous object regions by scene segmentation can be used to determine regions in the range map that can be enhanced by interpolation. This paper explores the use of scalar features derived from the spatial gray-level dependence matrix for texture segmentation. Thresholding of histograms of scalar texture features is done for several images to select scalar features which result in a meaningful segmentation of the images. Next, the selected scalar features are used with a neural net to automate the segmentation procedure. Back-propagation is used to train the feed forward neural network. The generalization of the network approach to subsequent images in the sequence is examined. It is shown that the use of multiple scalar features as input to the neural network result in a superior segmentation when compared with a single scalar feature. It is also shown that the scalar features, which are not useful individually, result in a good segmentation when used together. The methodology is applied to both indoor and outdoor images.

1 INTRODUCTION

Automatic guidance for aerospace vehicles can be accomplished in two stages. The first stage specifies a nominal vehicle path based on mission goals and a database containing digitized terrain geometry. The second stage specifies a modified path by constructing a three-dimensional model of the environment near the vehicle. The model of the environment near the vehicle is based on locally sensed information. For sensing, both active and passive sensors can be used.

Traditionally, the problem of ranging, using active sensors, has been studied extensively (ref. 1). These algorithms yield dense range maps (ref. 2). It is only recently that several techniques for ranging, using passive sensors such as electro-optic sensors, have appeared in the literature (refs. 3-9). It has been shown in references 3-5 that both stereo and motion algorithms provide a sparse set of ranges to discrete points in the image sequence as a function of azimuth and elevation. The problem of modeling dense range images, generated by active sensors, has been studied by several authors (refs. 2 and 10-12). This is usually achieved by fitting surfaces to the range data using polynomials, splines, and Delaunay triangles. In reference 12 the authors have also applied a surface fitting approach to depth data generated using a field based approach (ref. 9). The surface consistency constraint smoothes the depth values into regions where depth is unknown. From the examples in reference 12, it may be seen that surface fitting approaches do not work well on sparse range maps. The main reason for this is that the geometric relationships between the points of the range map are lost due to the discreteness of the range map. One of the ways of recovering this information is by application of problem-dependent constraints to cluster discrete ranges into few groups (ref. 13). The techniques for dense range maps may then be applied to each group to make the variation within the group continuous. The geometric relationships between the points may also be recovered by detecting the relationships in the image plane. One such way is to detect homogeneous regions by scene segmentation.

In the literature several methods have been proposed for scene segmentation. For example, the methods described in references 2 and 14-24, represent a few of the diverse approaches to scene segmentation. Some of these methods use texture features (refs. 19-24) for scene segmentation.

This paper reviews some of the texture based methods for scene segmentation and investigates the use of scalar texture features (ref. 22) for image segmentation. The selection of an appropriate set of texture features is the first step in segmentation; and towards this end, we examine the segmentation capability of individual features using thresholding techniques. The thresholding technique, while useful for individual features, is difficult to automate for multiple features. We approach the problem of segmenting the image using multiple features by training a feed-forward neural network (NN). The NN is trained using the method of back-propagation. Initially, the NN is trained with a single feature to ensure that its performance is comparable to that achieved by thresholding. It is shown that

the performance of the NN using several features is superior to the NN using a single feature. We also examine the capability of the NN to extrapolate by training it on one image and using it to segment another image. We present our results by applying them to both indoor and outdoor images. The NN approach to segmentation, though not completely satisfactory, shows great promise. We discuss how the results can be improved further. The paper is organized as follows: In section 2, different ways of characterizing the textural properties are described. In section 3, usefulness of each scalar feature is explored by evaluation of the segmentation achieved by thresholding of the histogram of the scalar feature. Neural network training and classification with a single scalar feature is described in section 4. Next, training and classification performance of a five input neural net using the five scalar features, found to be useful individually, is presented. Also, in section 4, results are presented for a five input neural net using a different set of scalar features. Some conclusions are drawn in section 5. Each scalar texture feature is defined in the appendix for completeness.

We thank Hien T. Tran of Analytical Mechanics Associates, Inc., for implementing the code and generating the results. We also thank Valerie Conti and R. Manmatha of the Computer Science Department, University of Massachusetts at Amherst, for providing the image in figure 33 and R. E. Suorsa of the Aircraft Guidance and Navigation branch at NASA Ames Research Center for providing figure 37.

2 TEXTURE

Gray scale images are characterized by pixels of varying intensity. Any image can be described by the nature of the distribution of the gray levels across the image. The properties of this distribution are usually described in terms of the first, second, and higher order statistics. First order statistics describe the pixel population in the image without regard to its spatial distribution, while the second order statistics take the spatial distribution into account. Two approaches are used to characterize this spatial distribution: (a) a stochastic model-based approach and (b) a data-driven approach. The model-based approach assumes that the image can be modeled in terms of two-dimensional random fields or time series. Several stochastic models are discussed in references 25 and 26).

The data-driven, or non-parametric approach, is based on characterizing the two-dimensional intensity distribution by different types and features of second order statistics. The conditional probability density function $f(i,j|d,\theta)$ represents the probability that two pixels separated by an interpixel distance d and orientation θ have intensities i and j. An estimate of the conditional probability density function, $c(i,j|d,\theta)$, is referred to as the gray-level co-occurrance matrix (GLCM) or as the spatial gray-level dependence matrix (SGLDM). SGLDM has been most widely used for classification of textures (refs. 19-20 and 22-24). SGLDM can be obtained by computing the two-dimensional histogram of the frequency of the joint occurrences of two pixels with a fixed displacement and orientation with respect to each other having intensities i and j respectively. A rotationally invariant SGLDM is computed by averaging the individual SGLDM for the angular directions.

For texture classification, either matrix features or scalar features are used. Many different approaches are available for texture classification using matrix features. Threshold selection based on the SGLDM is described in reference 19. In reference 20 the SGLDMs of four neighbors in the quad-tree are compared with a threshold for merging or splitting operations. Results using this technique are also presented in reference 21. A technique for image segmentation by detecting clusters in the SGLDM, which correspond to the regions and boundaries in the image, is described in reference 24. A maximum likelihood texture classifier using matrix and scalar features is examined in reference 23. In reference 27 segmentation is done by thresholding where the thresholds are selected by projecting the off-diagonal elements of the SGLDM onto the diagonal and treating the resulting vector as a histogram. Although, these methods are useful for segmentation, their storage requirements are high due to the use of matrix features. For example, 256×256 locations are needed to store a matrix feature for an image containing 256 gray-levels. These methods are also computationally intensive. The storage requirement and computational speed are the motivations for considering scalar features for image segmentation. However, it should be noted that many of the scalar features derived from the matrix features may not contain all the important texture information contained in the matrix features (ref. 28).

Several scalar features are derivable from the matrix features. For example, 14 scalar texture features based on the SGLDM are presented in reference 22.

For each of the scalar features their means and variance, computed by using the SGLDMs corresponding to the four directions, may be used for texture classification. Some scalar features derived from SGLDM, Fourier power spectrum, Gray level difference statistics, and Gray level run length statistics are described in references 28 and 29. Scalar texture features derived from the SGLDM may also be computed from sum and difference histograms (ref. 30). Compared to computing the full SGLDM, sum and difference histograms are fast computationally and require much less storage. Except for two scalar features, energy and entropy, all the scalar features can be obtained exactly by using the sum and difference histograms. Many of the methods such as references 20 and 23 can be used for classification using scalar features. Several other methods such as piecewise linear discriminant function method, min-max decision rule method reference 22, Fisher linear discriminant technique (ref. 29) can also be used for classification using scalar features.

Some of the scalar features relate to specific characteristics in the image such as, homogeneity, contrast, and organized structure. Other features characterize the complexity. Even though each scalar feature contains textural information, it is hard to identify which specific textural characteristic is represented by which feature. In this paper, we examine the classification ability of each scalar feature, derived from SGLDM, based on segmentation results of a laboratory image sequence and a natural scene. In most of the earlier work both with scalar and matrix features, synthetic textures, aerial images, and satellite images have been used. One of the objectives of this paper is to apply the scalar features to non-orthographic images with slant illumination. In the next section, we describe segmentation results obtained by thresholding the histograms of scalar features. These results show which of the scalar features are successful in classifying the image pixels into the desired categories.

3 FEATURE SELECTION

In this section, we present image segmentation results using the following scalar features: energy (f1), contrast (f2), correlation (f3), standard deviation (f4), local homogeneity (f5), entropy (f9), difference variance (f10), difference entropy (f11), difference average (f12), and mean (f13). Definitions of the various scalar features are given in appendix I. The information content in the scalar features, sum average (f6), sum variance (f7), and sum entropy (f8), is same as that in difference average (f12), difference variance (f10), and difference entropy (f11), respectively. Therefore, we will not consider f6, f7, and f8 in this study. Scalar features at every pixel of the image, shown in figure 1, are computed using a 17×17 window centered at the pixel. The image in figure 1 is the first in a series of 80 images. It consists of different objects like pencils, metal bracket, wooden block on a large optical table, and a textured wall in the background. If a particular scalar feature is a discriminant, it should be possible to threshold the histogram of the scalar feature into regions that correspond to the desired image segmentation. To investigate this, histograms for the scalar features are presented for the whole image and for rectangular regions shown in figure 2. The legends W, T, and O in figure 2 correspond to three categories: wall, table, and objects. The histogram of f1 for the whole image is shown in figure 3 and for the rectangular regions from the wall, table, and objects is shown in figure 4. In figure 4 the legends W, T, and O correspond to the regions shown in figure 2. The f1 histogram in figure 3 can be separated into three regions given by the thresholds $f1 < 0.008$, $0.008 \leq f1 < 0.03$, and $f1 \geq 0.03$. The histograms in figure 4 suggest that the image can be segmented into four categories using the thresholds: $f1 < 0.003$, $0.003 \leq f1 < 0.008$,

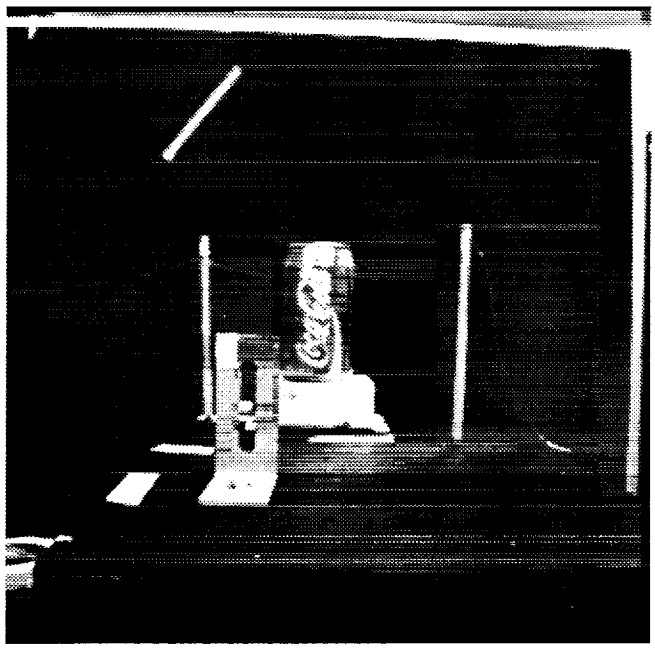

Figure 1. First lab image.

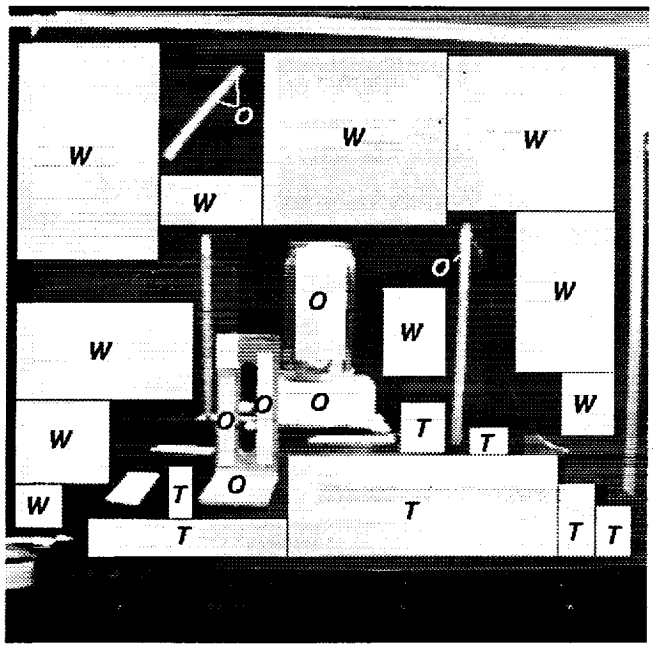

Figure 2. Regions in the first lab image.

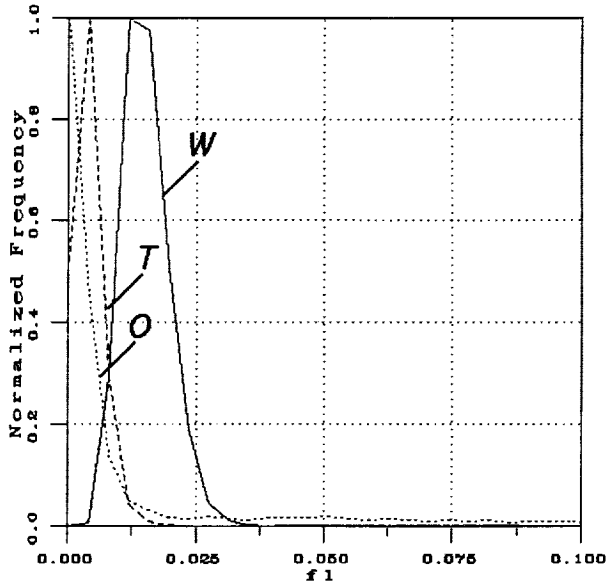

Figure 4. F1 histograms for the wall, table, and object regions in the first lab image.

$0.008 \leq f1 < 0.03$, and $f1 \geq 0.03$. The four thresholds result in the image segmentation shown in figure 5. In this case the lowest and the highest thresholds correspond to the object category. Image regions classified by the highest threshold are shown in white. In the absence of the truth data in figure 4, the image can only be segmented into three groups based on the f1 histogram of the whole image shown in figure 3. This would result in most of the object regions being classified as table regions. Although f1 feature can be used

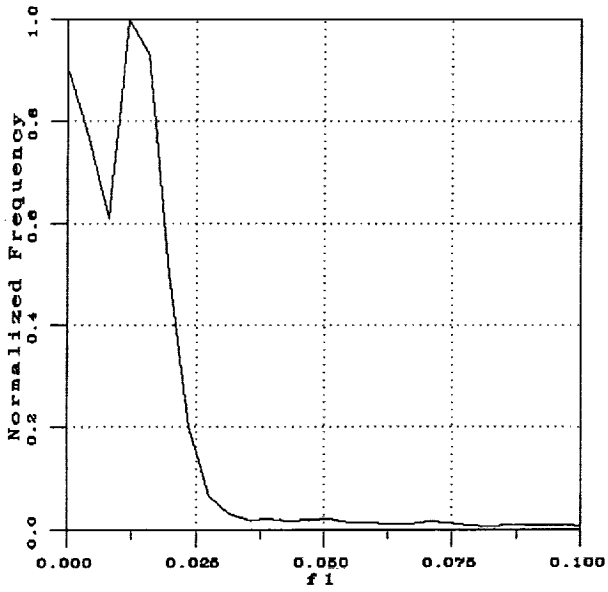

Figure 3. F1 histogram for the first lab image.

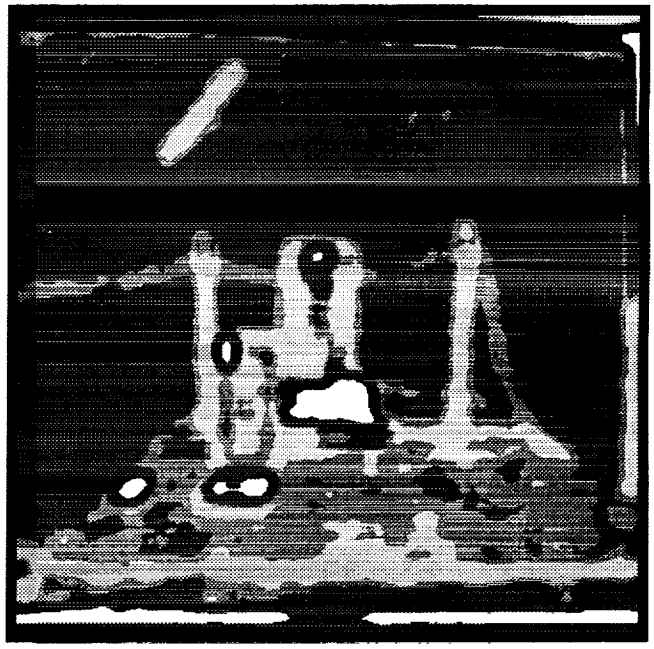

Figure 5. First lab image segmentation using f1 histogram.

for classifying the image in figure 1 into three categories, it is more suitable for a binary classification into wall and not-wall categories.

The f2 histogram for the whole image is shown in figure 6 and the histograms for the wall, table, and objects categories are shown in figure 7. The histograms in figures 6 and 7 suggest the segmentation thresholds: $f2 < 4$, $4 \leq f2 < 85$, and $f2 \geq 85$. Image segmentation with these thresholds is shown in figure 8. From the segmentation in figure 8, it may be seen that f2 correctly segments the image into the wall, table, and object categories.

Figure 9 shows the f3 histogram for the whole image. The f3 histograms for the wall, table, and object categories are presented in figure 10. From the histograms in figure 10, it may be seen that the wall and table categories are not separable. The three thresholds which partition the histogram in figure 9 are $f3 < 0.5$, $0.5 \leq f3 < 0.95$, and $f3 \geq 0.95$. This results in the segmentation shown in figure 11. From the segmentation in figure 11, it may be seen that f3 is useful for binary segmentation into object and not-object categories. The first threshold results in few pixels close to the right bottom corner (shown in white) being classified into a separate set.

The f4 histogram for the whole image is shown in figure 12 and the wall, table, and object histograms are given in figure 13. These histograms suggest four thresholds: $f4 < 1$, $1 \leq f4 < 4.5$, $4.5 \leq f4 < 16$, and $f4 \geq 16$. The segmented image corresponding to the four thresholds is given in figure 14. The segmentation in figure 14 is very similar to that in figure 8.

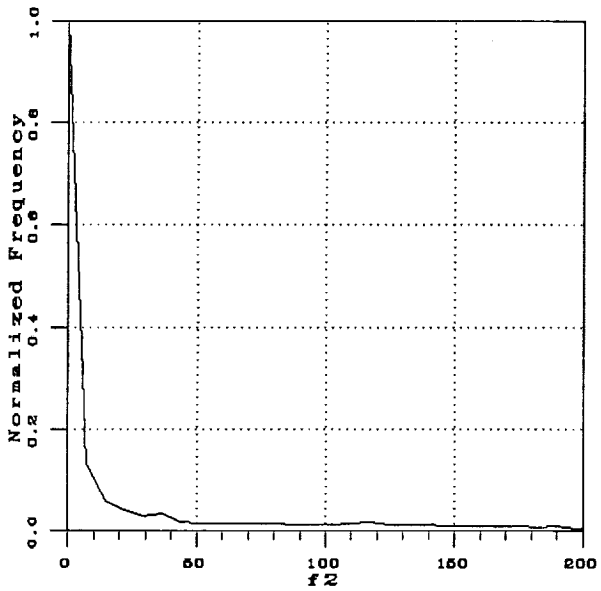

Figure 6. F2 histogram for the first lab image.

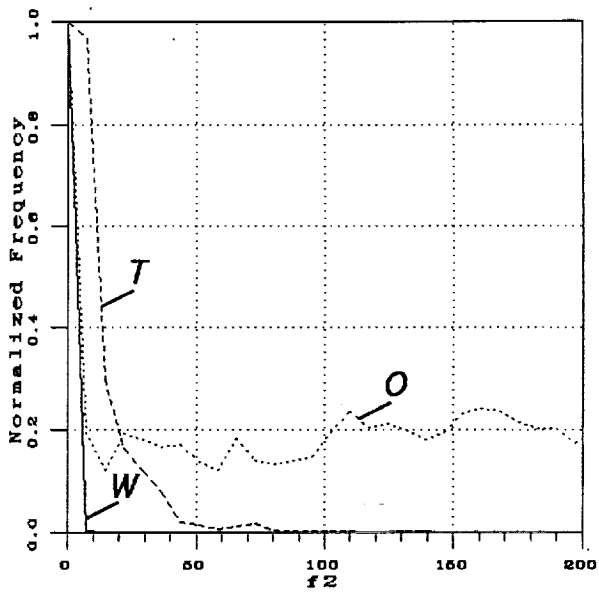

Figure 7. F2 histograms for the wall, table, and object regions in the first lab image.

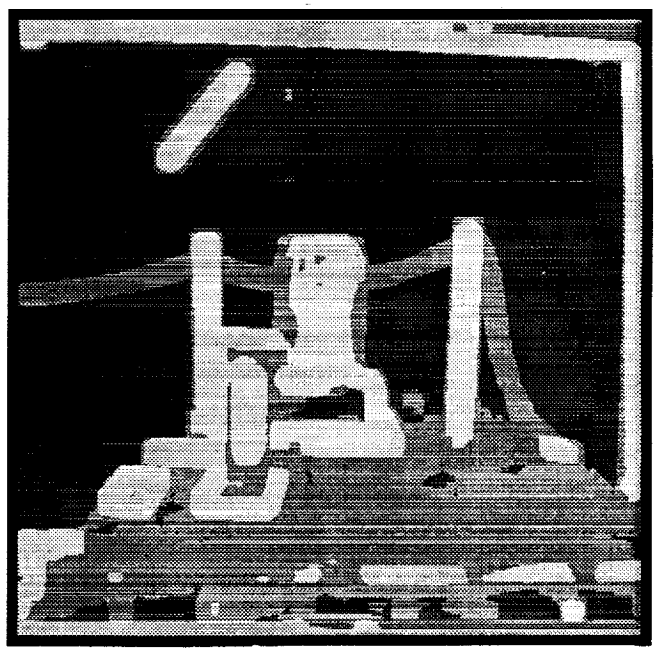

Figure 8. First lab image segmentation using f2 histogram.

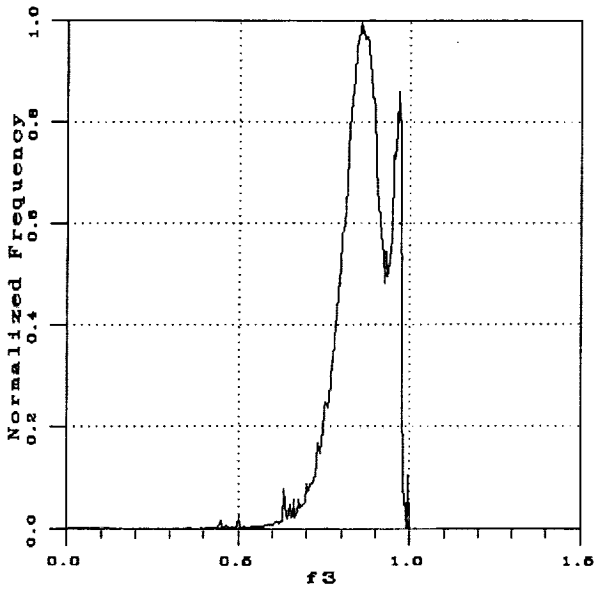

Figure 9. F3 histogram for the first lab image.

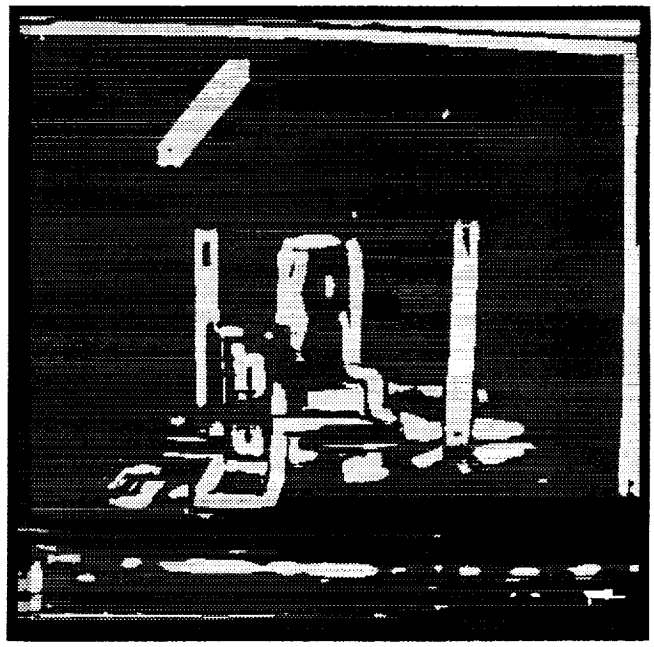

Figure 11. First lab image segmentation using f3 histogram.

The first threshold results in some of the object pixels (shown in white) being classified into a different category.

In the manner described above, four thresholds obtained by the f5 histograms, in figures 15 and 16, result in the image segmentation in figure 17. The f5 histogram for the whole image is shown in figure 15. The f5 histograms corresponding to the wall, table, and object regions are shown in figure 16. Segmentation thresholds obtained from these histograms are $f5 < 0.2$, $0.2 \leq f5 < 0.44$, $0.44 \leq f5 < 0.64$, and $f5 \geq 0.64$.

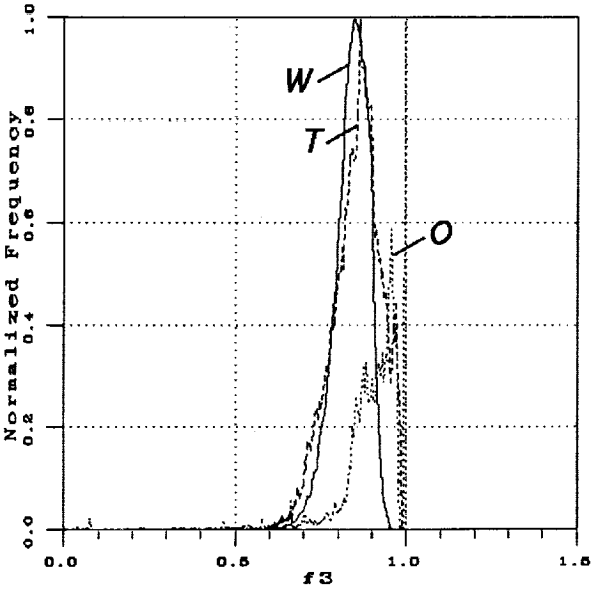

Figure 10. F3 histograms for the wall, table, and object regions in the first lab image.

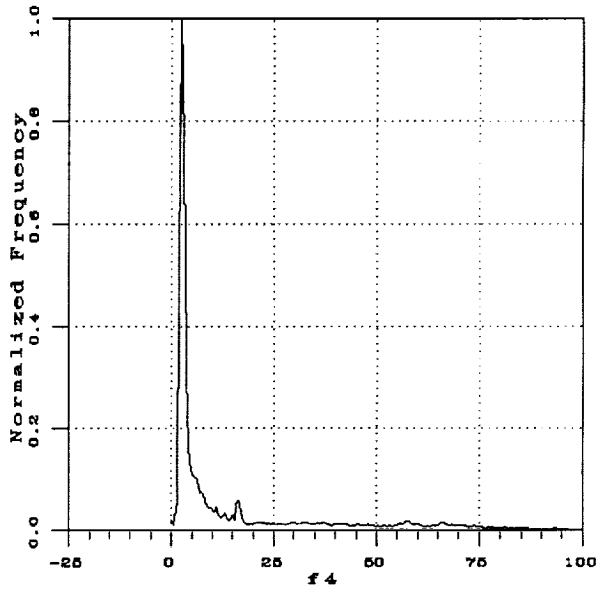

Figure 12. F4 histogram for the first lab image.

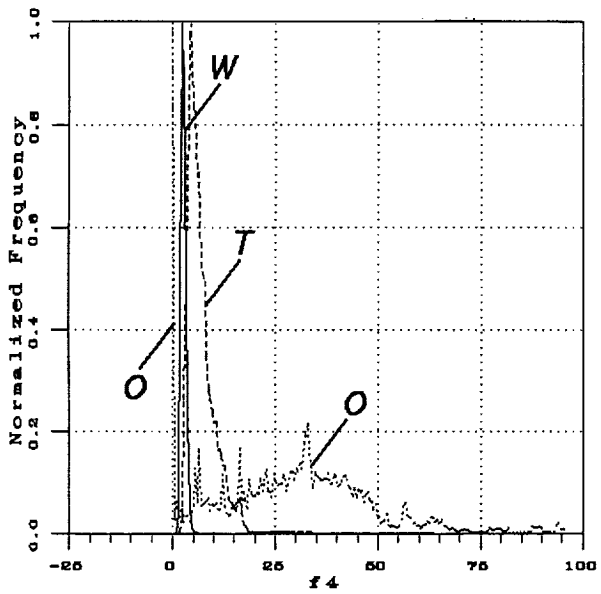

Figure 13. F4 histograms for the wall, table, and object regions in the first lab image.

The white regions in figure 17 correspond to the first threshold. The segmentation in figure 17 shows that f5 is useful for binary classification into wall and not-wall categories.

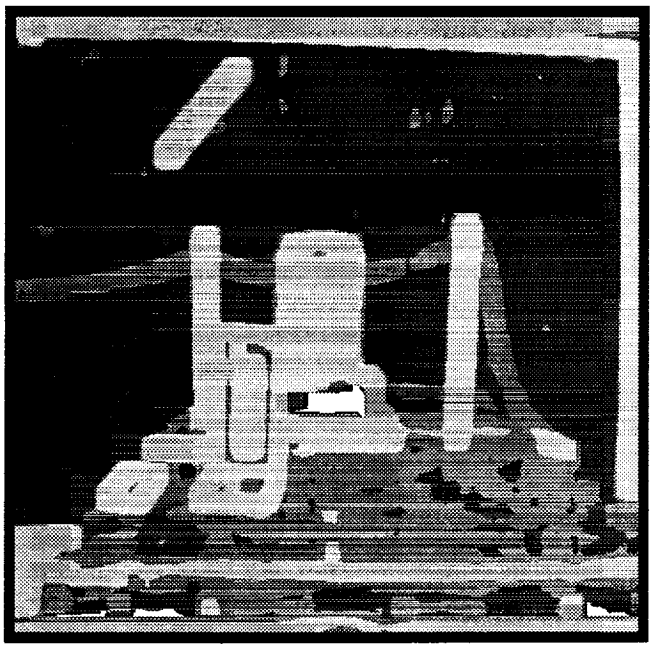

Figure 14. First lab image segmentation using f4 histogram.

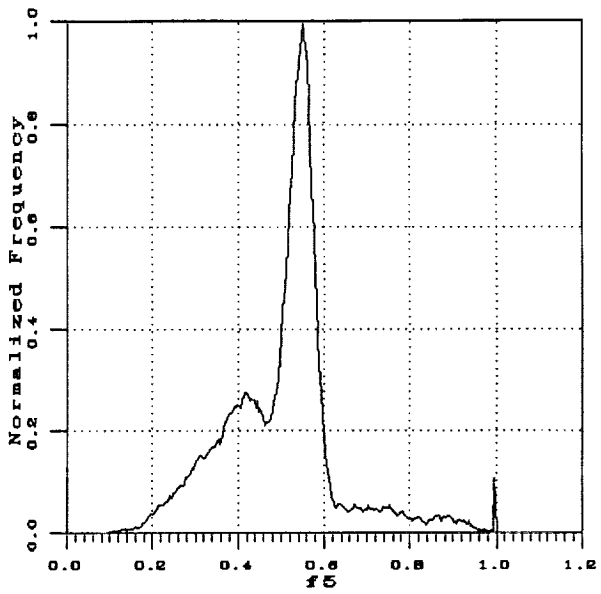

Figure 15. F5 histogram for the first lab image.

Histogram using f9 for the whole image is shown in figure 18. The f9 histograms for the wall, table, and object categories are shown in figure 19. Based on the histograms, the following thresholds, $f9 < 1.6$, $1.6 \leq f9 < 2.2$, $2.2 \leq f9 < 3$, and $f9 \geq 3$, result in the segmentation in figure 20. In figure 20, the white

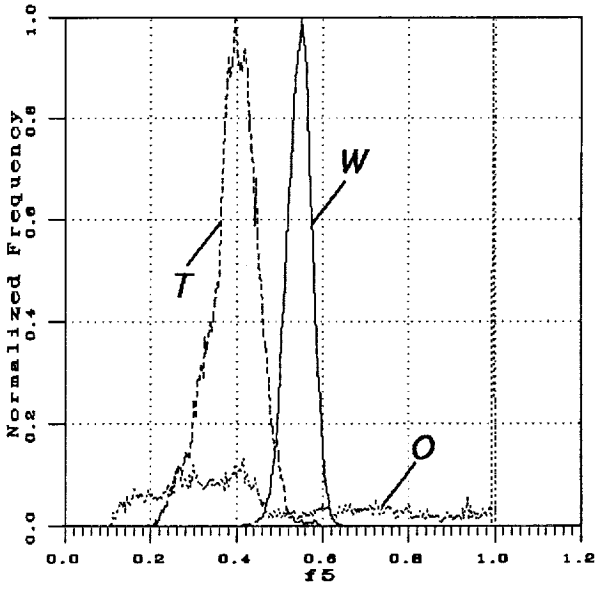

Figure 16. F5 histograms for the wall, table, and object regions in the first lab image.

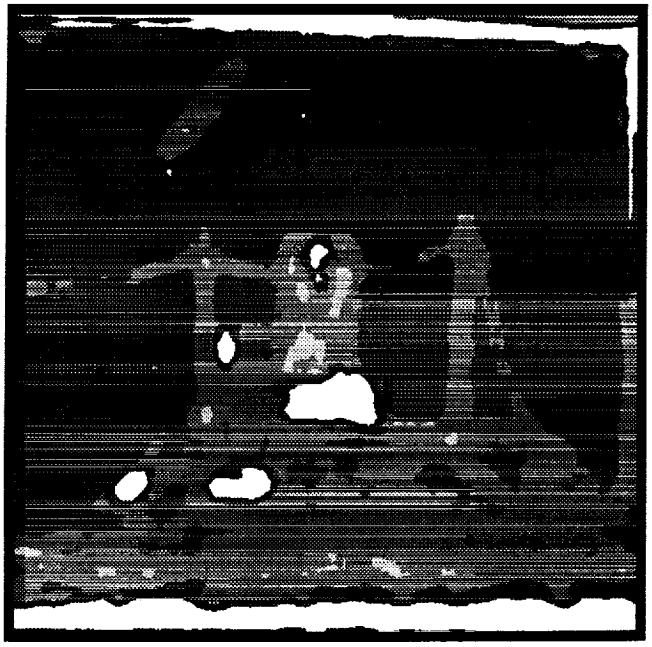

Figure 17. First lab image segmentation using f5 histogram.

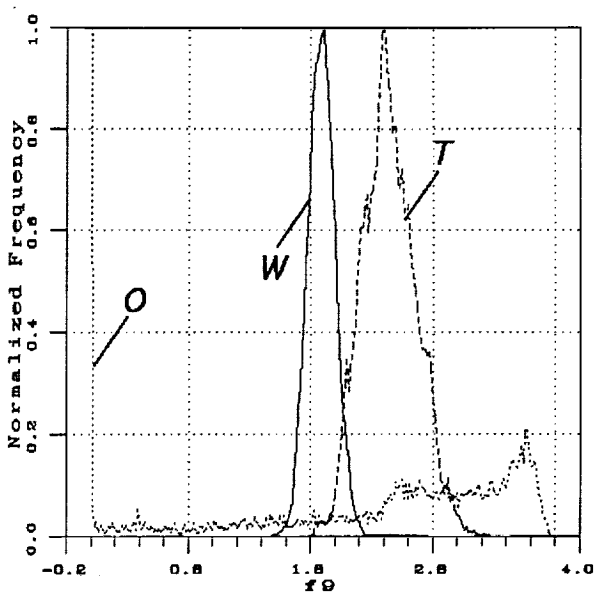

Figure 19. F9 histograms for the wall, table, and object regions in the first lab image.

regions correspond to the first threshold. From the segmentation results it may be seen that most of the wall, table, and object pixels are classified correctly by the f9 feature.

In figure 21 the f10 histogram for the whole image is given, and in figure 22 the f10 histograms for the wall, table, and objects are given. The segmentation resulting from the histogram thresholds, $f10 < 0.6$, $0.6 \leq f10 < 1.3$, $1.3 \leq f10 < 6$, and $f10 \geq 6$, is given in figure 23. The segmentation result in figure 23

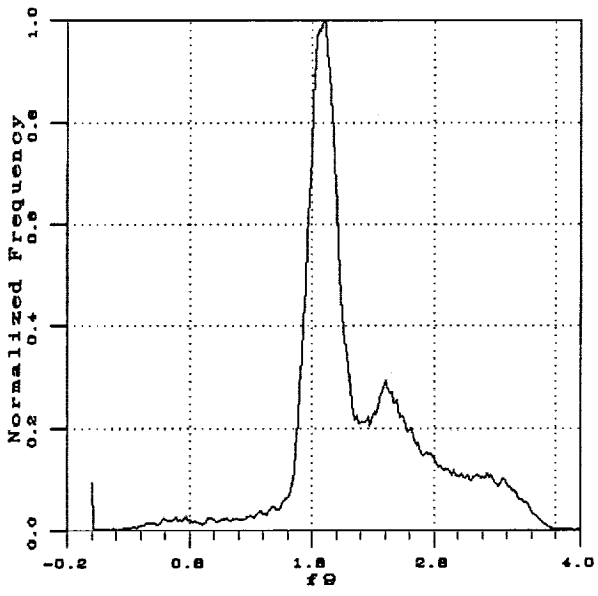

Figure 18. F9 histogram for the first lab image.

Figure 20. First lab image segmentation using f9 histogram.

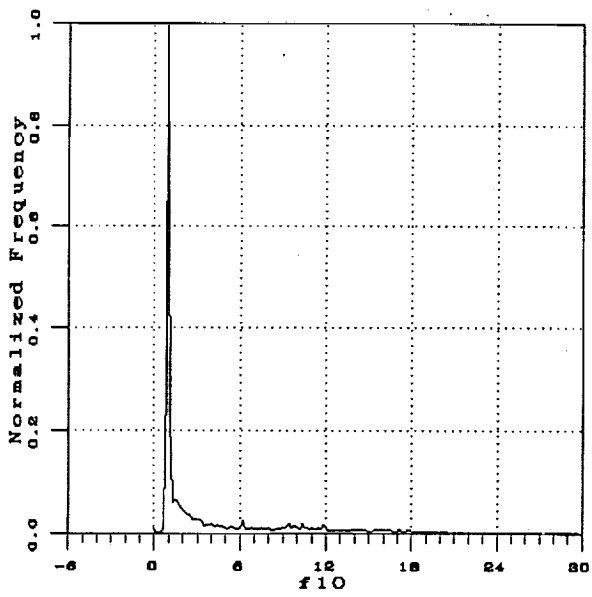

Figure 21. F10 histogram for the first lab image.

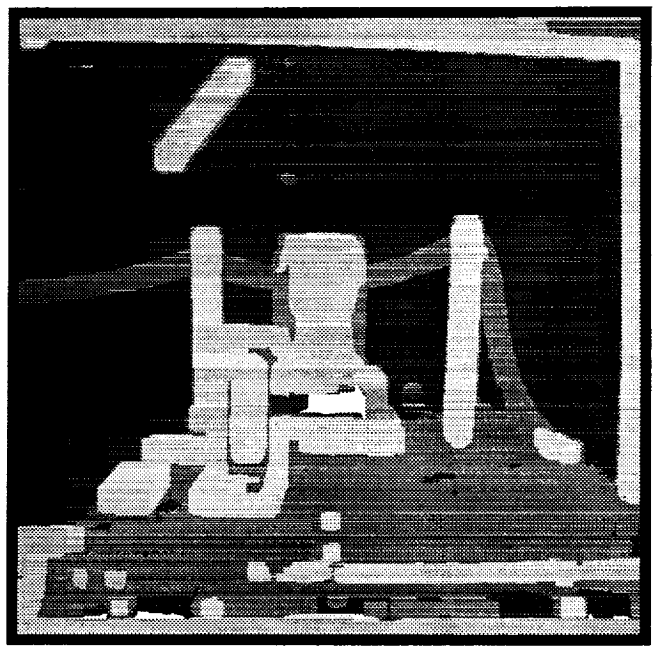

Figure 23. First lab image segmentation using f10 histogram.

is similar to that in figure 14. Image pixels classified by the first threshold are shown in white.

Thresholds derived from the f11 histograms presented in figures 24 and 25 are $f11 < 0.5$, $0.5 \leq f11 < 0.7$, $0.7 \leq f11 < 1$, and $f11 \geq 1$. Here, the f11 histogram for the complete image is shown in figure 24 and the wall, table, and object histograms are shown in figure 25. Based on the f11 thresholds, the image segmentation is presented in figure 26. Regions shown in white in figure 26 correspond to the first

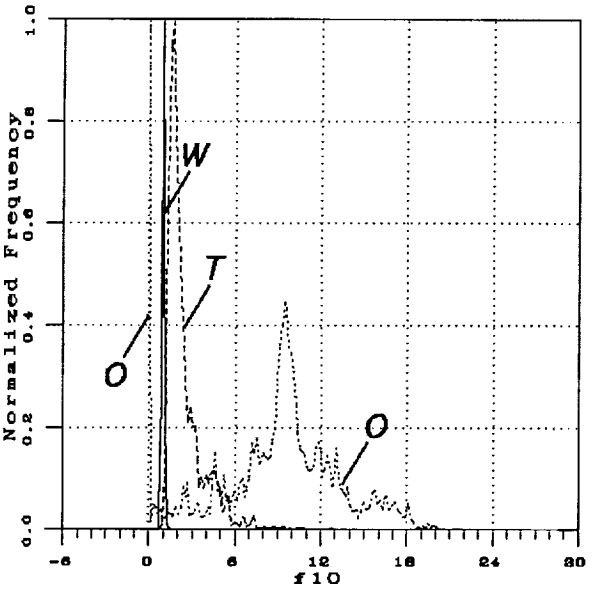

Figure 22. F10 histograms for the wall, table, and object regions in the first lab image.

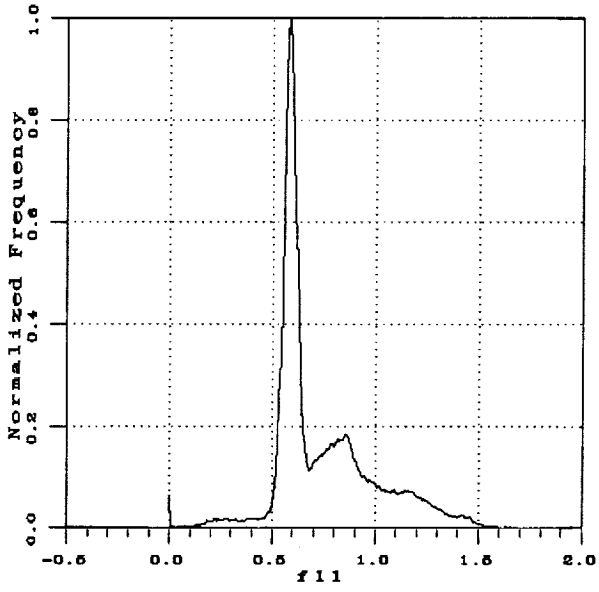

Figure 24. F11 histogram for the first lab image.

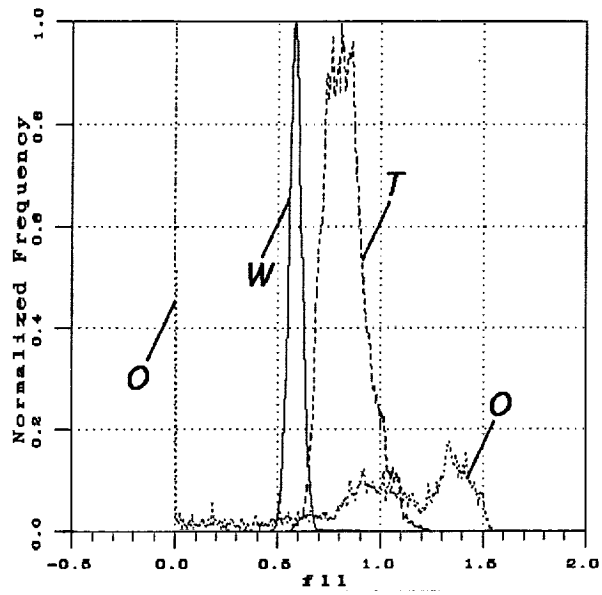

Figure 25. F11 histograms for the wall, table, and object regions in the first lab image.

threshold. The segmentation result is very similar to that in figure 20.

The f12 histogram for the whole image in figure 27 and the histograms for wall, table, and object categories in figure 28, result in the thresholds $f12 < 0.8$, $0.8 \leq f12 < 1.5$, $1.5 \leq f12 < 3.8$, and $f12 \geq 3.8$. These thresholds lead to the segmentation in figure 29. This segmentation result is similar to those in figures 14 and 23. The white regions in figure 29 correspond to the lowest threshold.

Histograms using f13 for the whole image is given in figure 30. Similar histograms for wall, table, and object regions are given in figure 31. The following thresholds, $f13 < 38$, $38 \leq f13 < 110$, and $f13 \geq 110$, partition the image into three groups shown in figure 32. The first threshold corresponds to the white region in figure 32. The segmentation in figure 32 suggests that f13 can partition the image into object and non-object categories. It may be seen that parts of the table are classified as objects due to the fact that f13 is directly effected by the scene illumination.

In summary, the following may be said regarding the classification performance of the 10 scalar features examined on the image in figure 1: f1, f3, and f5 are useful for binary segmentation; f2, f4, f9, f10, f11, and f12 are useful for segmentation into the desired categories. The features f4, f10, and f12 result in similar segmentations. Also, f9 and f11 result in similar segmentations. The scalar feature f13 is tone dependent and therefore, may not be very useful for images with gradual tonal variation caused by illumination.

To verify that the scalar features found to be good discriminants for the laboratory image in figure 1 are also good discriminants for other images, we consider

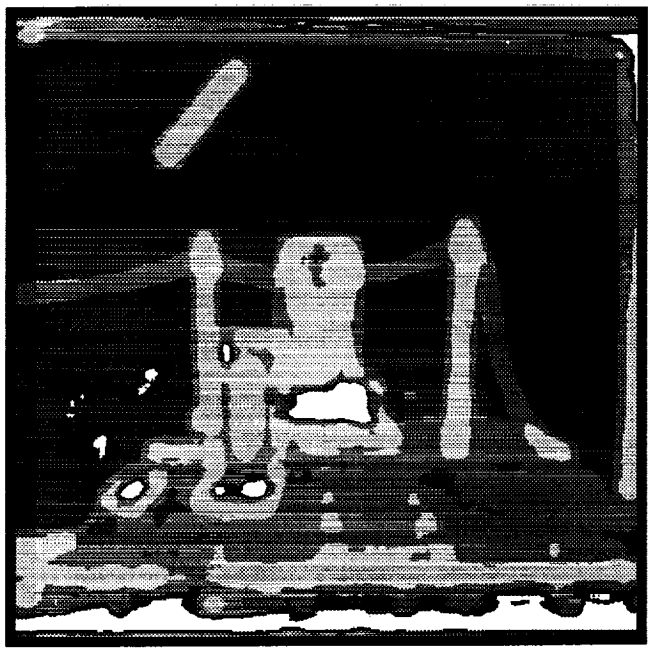

Figure 26. First lab image segmentation using f11 histogram.

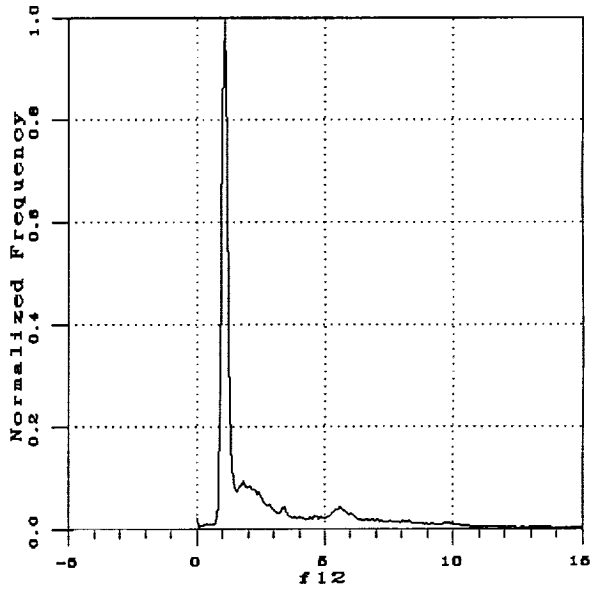

Figure 27. F12 histogram for the first lab image.

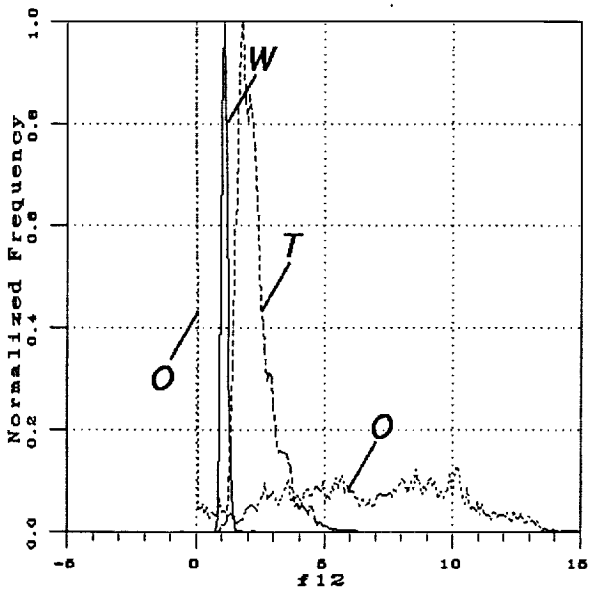

Figure 28. F12 histograms for the wall, table, and object regions in the first lab image.

a sequence of 30 images acquired by the Autonomous Land Vehicle (ALV) in the area surrounding the Martin

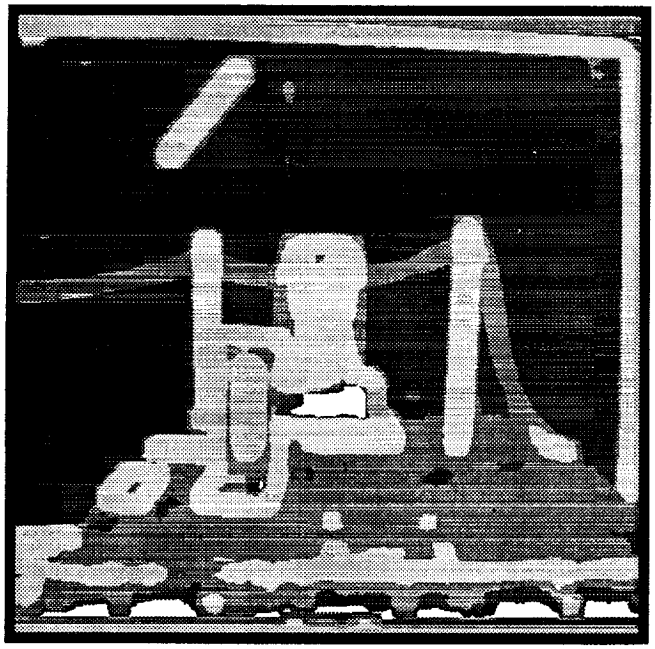

Figure 29. First lab image segmentation using f12 histogram.

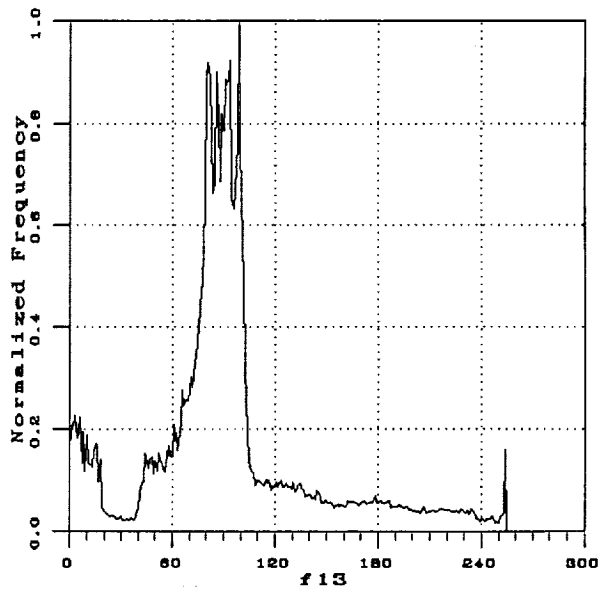

Figure 30. F13 histogram for the first lab image.

Marietta plant in Denver. The details of the image acquisition method and decription of the motion data associated with the images are available from the University of Massachussetts (UMASS) at Ahmerst (ref. 31). Thresholds were obtained for the histograms of scalar

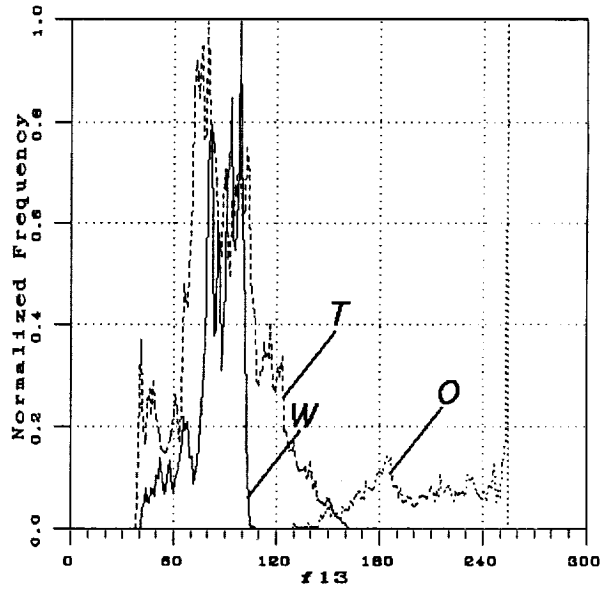

Figure 31. F13 histograms for the wall, table, and object regions in the first lab image.

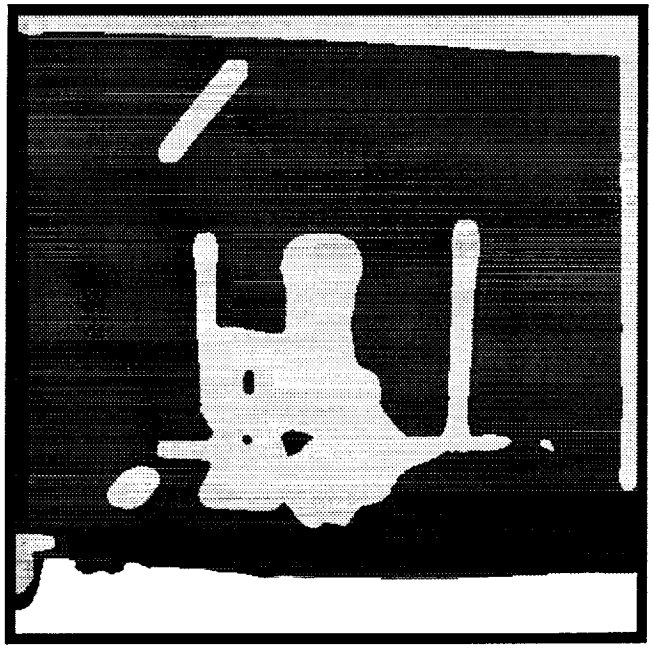

Figure 32. First lab image segmentation using f13 histogram.

features computed on the UMASS image shown in figure 33. The thresholds obtained for the various scalar features are summarized in table 1. Here, the labels M, G, and S represent three categories namely, mountain, ground, and sky. From the table it may be seen that f1, f4, f5, f9, f10, f11, and f12 result in a meaningful

Figure 33. UMASS Image.

Table 1. Thresholds for scalar features (UMASS image)

Features	M	G	S
f1	0-0.001	0.001-0.005	0.005-0.2
f2	70-300	0-70 300-15236	0-70 300-15236
f3	0.6-1.0	0.6-1.0	0-0.6
f4	10-23	5-10 23-126	0-5
f5	0-0.2	0.2-0.28 0.38-1	0.28-0.38
f9	3.1-3.6	0-1.8 2.4-3.1 3.6-4	1.8-2.4
f10	4-10	2.2-4 10-106	0-2.2
f11	1.2-1.5	0.9-1.2 1.5-1.8	0-0.9
f12	6-11.5	3-6 11.5-65	0-3
f13	60-100	0-60 100-255	0-60 100-255

segmentation of the image in figure 33 into the mountain, ground, and sky categories. The feature f2 separates mountain from not-mountain, f3 separates sky from not-sky, and f13 separates mountain from not-mountain.

The segmentation resulting from thresholding the histogram of f9 feature is shown in figure 34. This result is representative of the segmentations obtained by using f1, f4, f5, f9, f10, f11, and f12. Binary segmentation obtained by using f2 and f3 are shown in figures 35 and 36.

The set of scalar features which result in an acceptable segmentation of both, the image in figure 1 and figure 33 are f4, f9, f10, f11, and f12. The experience with the two images, used in this paper, indicates that it may be possible to segment other images using

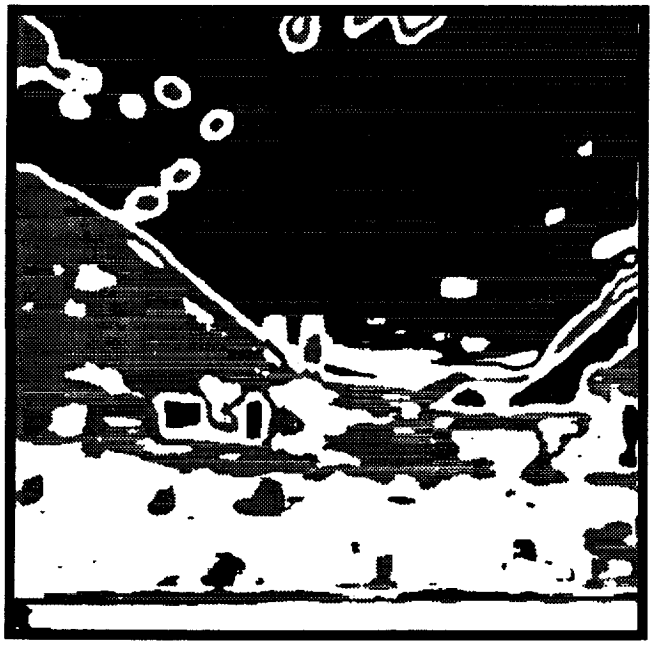

Figure 34. UMASS image segmentation using f9 histogram.

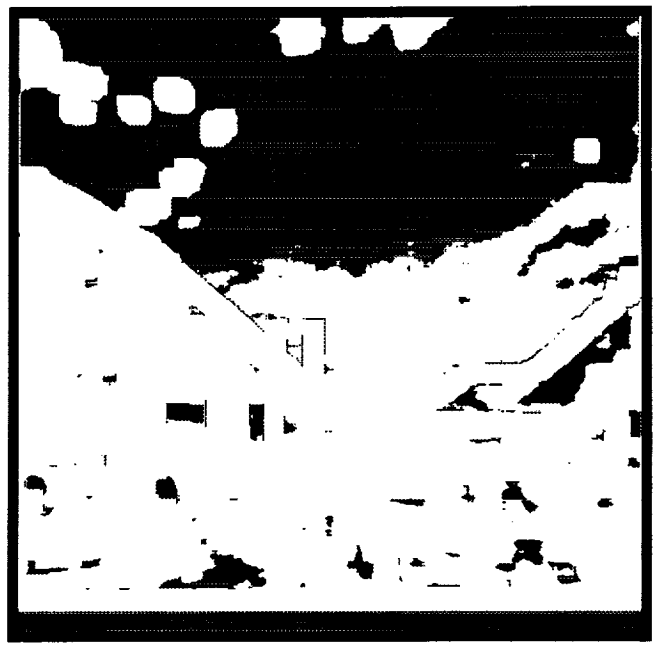

Figure 36. UMASS image segmentation using f3 histogram.

these features individually. So far we have not examined the classification achievable by using several features together. It is quite possible that several features together may improve the segmentation. We explore this idea with a neural network, described later. In the next section we describe a neural network approach (ref. 32) for supervised classification using single and multiple scalar features.

Figure 35. UMASS image segmentation using f2 histogram.

4 SEGMENTATION WITH NEURAL NETWORKS

In the previous section we have established that the scalar features that show most promise for image segmentation are f4, f9, f10, f11, and f12. These five features may be used individually or together to train a multilayer neural net (NN) using back propagation (ref. 32). Figure 37 shows an example of a generic two layer feed-forward NN. As shown in the figure, the input nodes are connected to the nodes in the hidden layer via weights, biases, summing junctions, and sigmoids. The nodes in the hidden layer are also connected to the output nodes in a similar way.

In this section we will first illustrate NN training and classification with a single scalar feature and then with five scalar features. The reason for using a single feature is that some of the results are easier to relate to the thresholds obtained from the histograms of the scalar features, discussed earlier, and to validate that the chosen scalar features are texture discriminants.

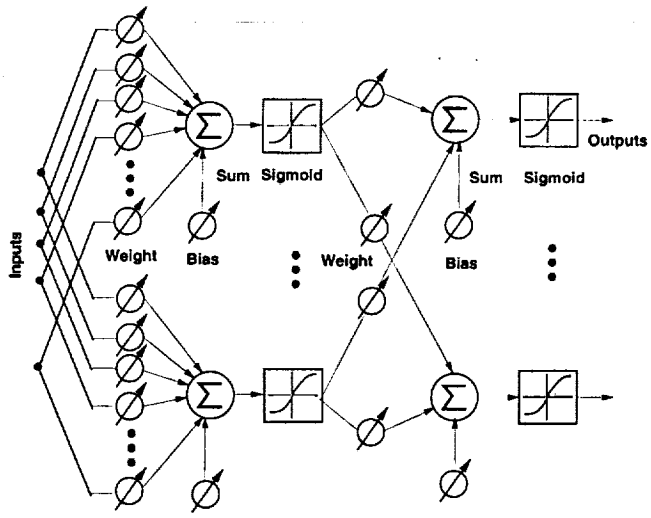

Figure 37. A two layer feed-forward neural network.

4.1 Training with Single Inputs

A feed-forward multilayer NN with one node in the input layer, four nodes in the hidden layer and three nodes in the output layer was used for classification. The NN was trained with f9 using back propagation. For training and NN performance evaluation, several rectangular regions from the wall, table, and objects were chosen. These regions are shown in figure 2. In the figure, the labels W, T, and O correspond to wall, table, and objects. From each of these regions 80 pixels were randomly chosen for training of the NN. The performance of the trained NN for the training samples is summarized in table 2. The legends W, T, O, and U

Table 2. Training performace using f9 for the first lab image

	W	T	O	U
W	80			
T		77		3
O			80	

represent wall, table, objects, and unknown. The table shows that all the wall and the object samples were classified correctly. Out of 80 table samples, 77 were classified correctly and three samples could not be classified at all. The U (unknown category) is that in which all three outputs of the NN were below a threshold of 0.6. The three NN outputs for the 240 pixels (80 from wall, 80 from table, and 80 from objects) are shown in figure 38. It may be noted that for each pixel all three outputs have a value and therefore, each curve consists of 240 points. If a classification threshold of 0.6 is chosen, the thresholds on f9 obtained from figure 38 are $f9 < 1.6$, $1.6 \leq f9 < 2.2$, $2.2 \leq f9 < 3.0$, and $f9 \geq 3.0$. It may be seen that these thresholds are same as those which partition the f9 histogram in figure 18.

The convergence characteristics of the NN are shown in figure 39. The 23 weights of the NN converged in 632 cycles. Figure 39 shows the error, E, and the probability of correct classification, PCC, as a function of a number of cycles. It may be noted that E is a feature of the fit error and not classification error. PCC is defined as the ratio of the number of samples correctly classified to the number of samples input to NN.

For evaluation of the performance of the trained NN for all samples within the rectangular regions, shown in figure 2, the f9 values for each pixel in the image in figure 1, were input to the trained NN for classification into wall, table, object, or unknown categories. The resulting segmentation is shown in figure 40. The white pixels in figure 40 correspond to the unknown category. It is interesting to note that the white pixels are usually along the edges separating two categories. The segmentation result presented here is very similar to that shown in figure 20 except that many of the

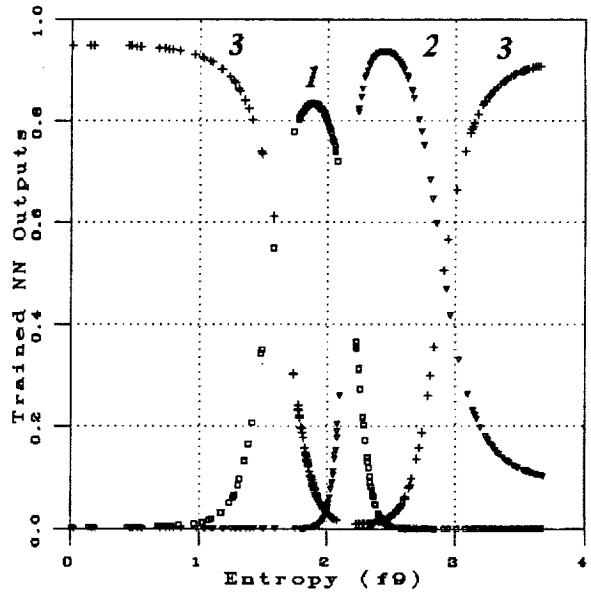

Figure 38. Outputs of NN using f9 for the first lab image.

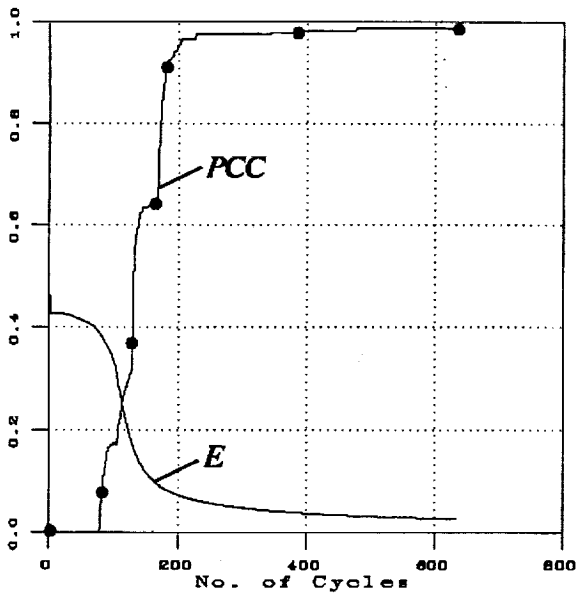

Figure 39. Convergence of NN using f9 for the first lab image.

regions shown in white in figure 20 are classified as objects. The $f9$ values of these regions are between 0 and 1.6.

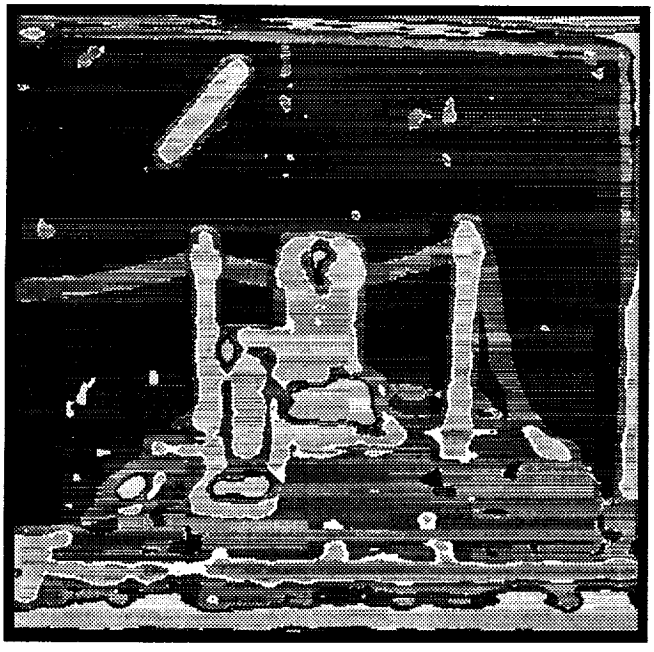

Figure 40. First lab image segmentation with NN using f9.

The NN performance for all the pixels in the rectangular regions in figure 2 is summarized in table 3. The rectangular regions in figure 2 contain 92615 wall pixels, 27708 table pixels, and 8456 object pixels. From the table it may be seen that 98% of the wall pixels, 87.2% of the table pixels and 63.4% of the object pixels were classified correctly.

Table 3. Correct classification using f9 for all pixels of the first lab image

	W	T	O	U
W	90817	1200	195	403
T	2071	24152	335	1150
O	656	1923	5364	513

4.2 Generalization

To evaluate the ability of the NN to classify pixels of another image in the sequence, the f9 value of each pixel in the 80th image was input to the trained NN. The resulting segmentation is shown in figure 41. By comparing figure 41 to figure 40, it may be seen that the NN trained on the first image results in a meaningful segmentation of the 80th image in the sequence. This

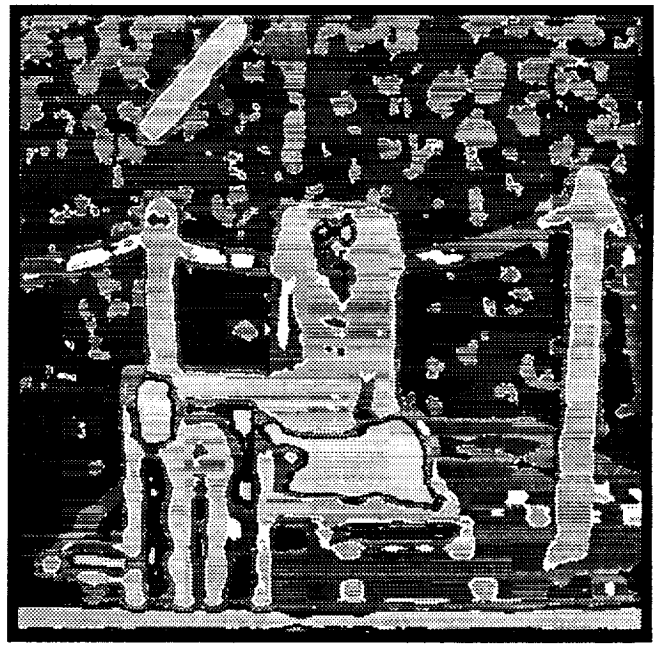

Figure 41. 80th lab image segmentation with NN using f9.

example illustrates the ability of the NN, using scalar texture features, to generalize to other images in the sequence. As the camera is brought closer to scene, the texture properties of the objects in the scene may change and therefore, the NN will have to be re-trained beyond a certain number of images.

4.3 Performance with Single Inputs

As described above for f9, NNs with one node in the input layer, four nodes in the hidden layer and three nodes in the output layer were trained with f4, f10, f11, and f12 to classify every pixel of the image in figure 1 as wall, table, or object pixel. The performance of the NNs for the rectangular regions in figure 2 is summarized in table 4. Table 4 shows that classification

Table 4. Classification of first lab image regions using different scalar features

Features	W	T	O
f4	97.4%	71%	71.2%
f9	98%	87.2%	63.4%
f10	98.3%	89.8%	75.4%
f11	97.8%	79.5%	55.6%
f12	99%	56.9%	70.5%

performance with f10 is better than with f4, f9, f11, and f12 for the image in figure 1. All the features are good for classification of wall pixels. The scalar features f11 and f12 are not good discriminants for object and table pixels, respectively. The ability of different scalar features to separate the various categories in an image varies as shown in table 4 therefore, it may be possible to use several scalar features together to achieve a better image segmentation. We explore this idea next.

4.4 Training with Multiple Inputs

A NN with five nodes in the input layer, four nodes in the hidden layer, and three nodes in the output layer was trained with the normalized values of f4, f9, f10, f11, and f12 using back propagation for classification of the pixels in figure 1 into wall, table, and object pixels. For normalization, the value of the scalar feature at every pixel in the image was divided by the maximum value of that scalar feature in the image. This makes the NN unbiased to a particular scalar feature. To train the NN, 80 samples from the wall, 80 samples from the table, and 80 samples from the objects were randomly chosen from the rectangular regions in figure 2. The 39 weights of the NN converged in 209 cycles. The convergence characteristics of the NN are shown in figure 42. In figure 42, E is the fit error, and PCC is the probability of correct classification.

For performance evaluation of the trained NN for pixels in figure 2, f4, f9, f10, f11, and f12 values for each pixel in figure 1 were input to the trained NN for classification into wall, table, object, or unknown categories. The resulting segmentation is shown in figure 43. For the rectangular regions in figure 2, the segmentation in figure 43 shows that 98.3% of the wall pixels, 89.9% of the table pixels, and 93.3% of the object pixels were classified correctly. On comparing these with those in table 4, it may be seen that the NN using the five scalar features generally performs better than the NNs using a single scalar feature. Most of the improvement results from increased accuracy in classifying object pixels.

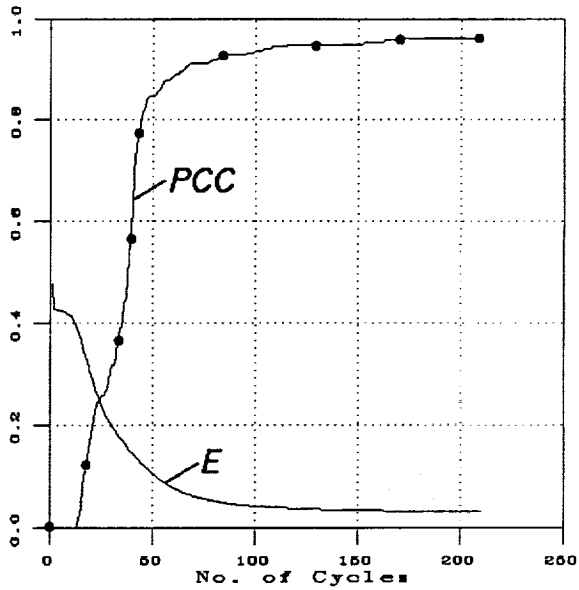

Figure 42. Convergence of NN using f4, f9, f10, f11, and f12 for the first lab image.

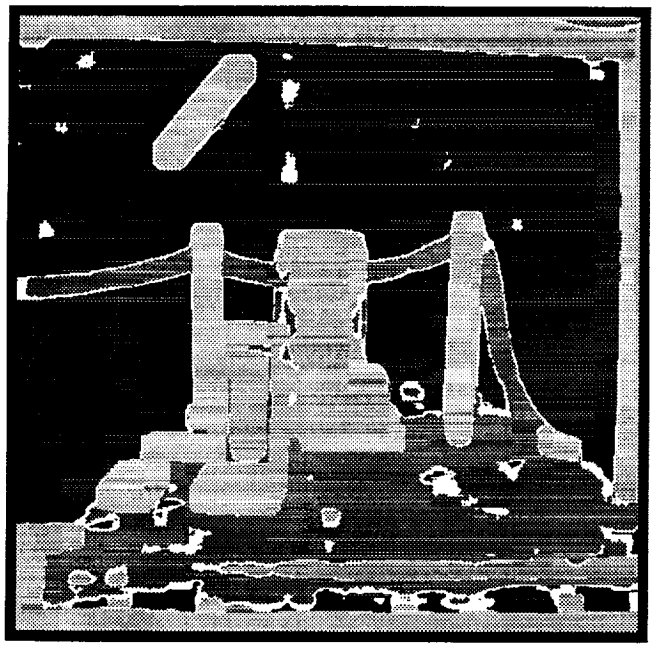

Figure 43. First lab image segmentation with NN using f4, f9, f10, f11, and f12.

4.5 Generalization with Multiple Inputs

Segmentation result of the 80th image with the NN trained on the first image is shown in figure 44. It is interesting to see that the wire is classified as an object in this image compared to it being classified as table in figure 41. By comparing figure 44 to figure 41, it can be said that the classification in figure 44 is superior to that in figure 41.

4.6 Single Input Training for UMASS Image

In order to examine if the five scalar features discussed above would work for other images, NNs were trained using a single scalar feature and the five features together to classify regions in the image in figure 33 as mountain, ground, and sky regions. For individual scalar features, NNs with one node in the input layer, four nodes in the hidden layer, and three nodes in the output layer were used. For training, value of the scalar features for 120 points from the mountain, 120 points from the ground, and 120 points from the sky were used. The performance of the NNs with different scalar features was evaluated on how many of the 14441 mountain pixels, 55524 ground pixels, and 99376 sky pixels were classified correctly. Both for training and for performance evaluation, pixels were chosen from the rectangular regions with legends S, M, and G, shown in figure 45. In this figure, regions from the sky, mountain, and the ground are marked with labels S, M, and G, respectively. Table 5 summarizes the results for the five scalar features. From

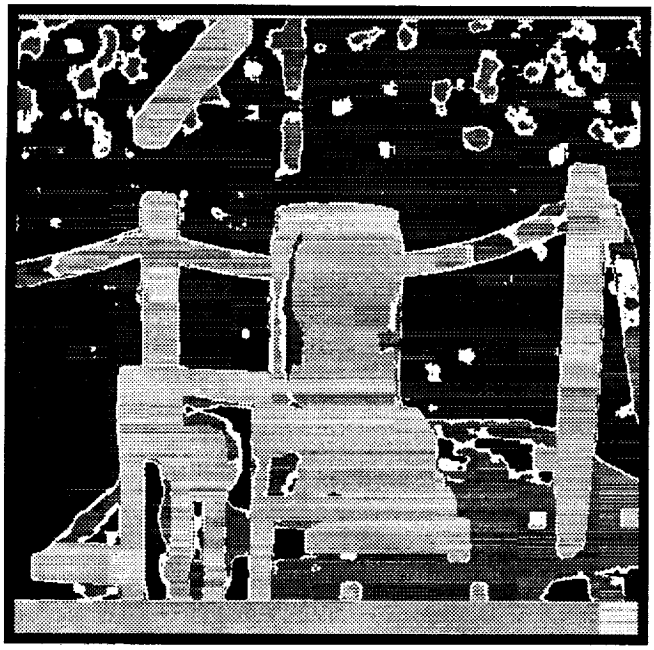

Figure 44. 80th lab image segmentation with NN using f4, f9, f10, f11, and f12.

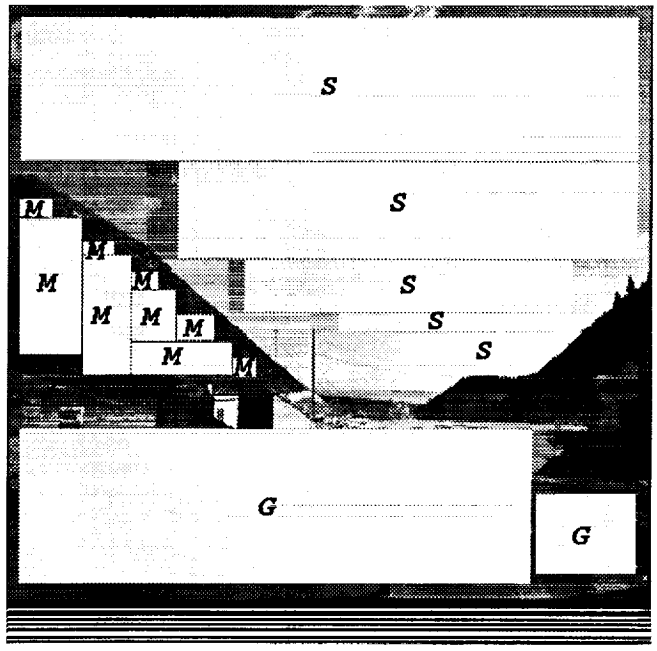

Figure 45. Sky, mountain, and ground regions in the UMASS image.

Table 5. Classification of UMASS image regions using different scalar features

Feature	M	G	S
f4	91.5%	67.5%	87.1%
f9	96.2%	66%	86.4%
f10	97%	68.1%	89.1%
f11	98.1%	64.4%	89.6%
f12	99%	65%	90.6%

the table it may be seen that the five scalar features perform similarly. The performace of f12 is somewhat better than others.

4.7 Multiple Input Training for UMASS Image

A NN with five nodes in the input layer, four nodes in the hidden layer, and three nodes in the output layer was trained on the normalized values of f4, f9, f10, f11, and f12 using back propagation to classify every pixel of the image in figure 33 into mountain, ground, or sky pixel. The 39 weights of this NN converged in 700 cycles. The convergence characteristics of the NN is shown in figure 46. The converged NN classified 98.1% of the 14441 mountain pixels, 89.1% of the 55524 ground pixels, and 89.4% of the 99376 sky pixels. By comparing these numbers to those in table 4, it may be seen that classification of the ground pixels improves considerably. The classification of the mountain and sky pixels is comparable to that achieved by a single scalar feature. The segmentation achieved by the classification of every pixel in figure 33 by the NN is shown in figure 47.

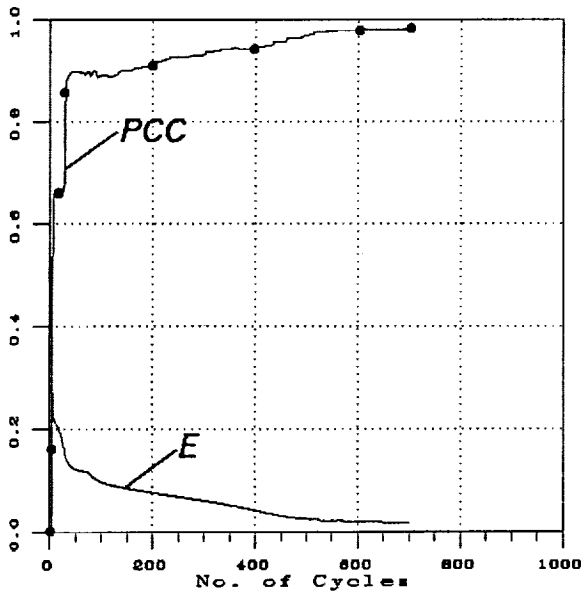

Figure 46. Convergence of NN using f4, f9, f10, f11, and f12 for the UMASS image.

4.8 Performance with Discarded Features

So far we have not considered the other five scalar features namely, f1, f2, f3, f5, and f13, together. During the feature selection process we had discarded these scalar features based on their inability to segment the images in figure 1 and figure 33 into the desired regions. To evaluate if they are useful together, a NN with five nodes in the input layer, four nodes in the hidden layer, and three nodes in the output layer was trained on the normalized values of f1, f2, f3, f5, and f13 using back propagation to classify every pixel of the image in figure 1 as wall, table, or object pixel. The 39 weights of this NN converged in 435 cycles. The convergence characteristics of the NN is shown in figure 48 and the resulting segmentation is shown in figure 49. The trained NN classified 98% of the wall pixels, 90% of the table pixels, and 99.6% of the object pixels correctly. The classification performance

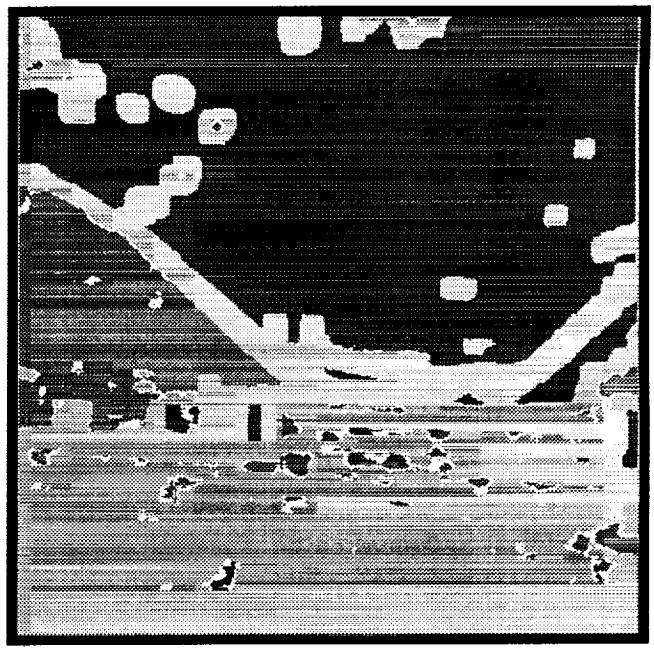

Figure 47. UMASS image segmentation with NN using f4, f9, f10, f11, and f12.

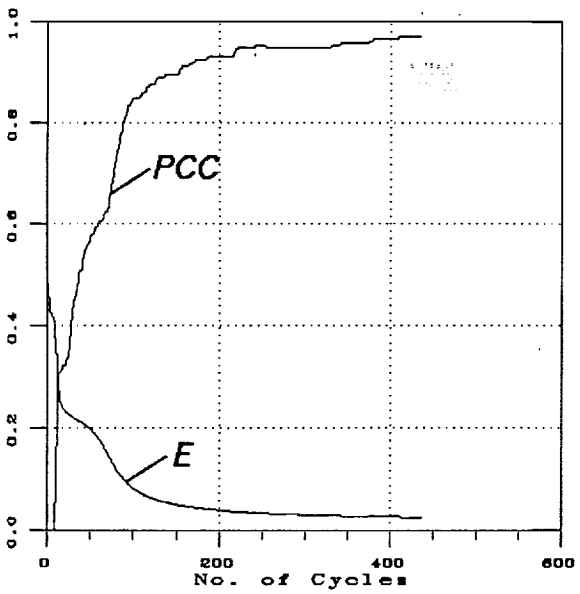

Figure 48. Convergence of NN using f1, f2, f3, f5, and f13 for the first lab image.

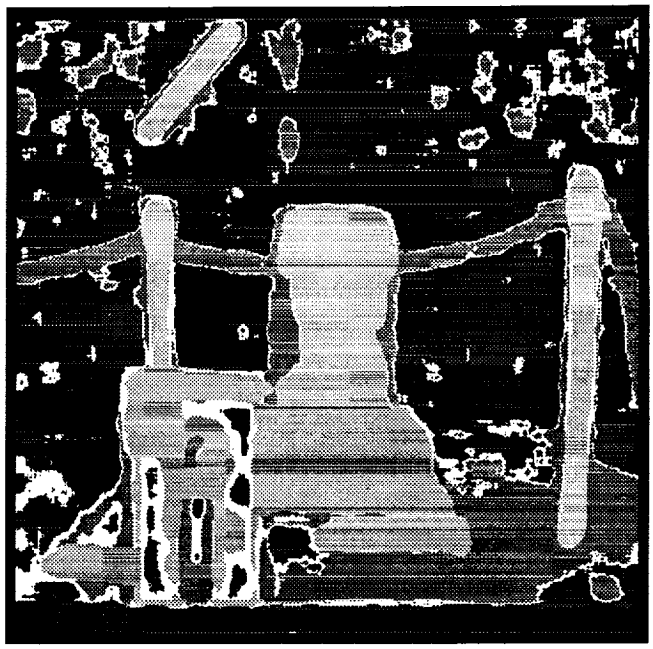

Figure 50. 80th lab image segmentation with NN using f1, f2, f3, f5, and f13.

is comparable to that achieved by using the scalar features f4, f9, f10, f11, and f12 together where, 98.3% of the wall pixels, 89.9% of the table pixels, and 93.3% of the object pixels were correctly classified. The classification performance on the 80th image is shown in figure 50. This result is also comparable to that shown in figure 44.

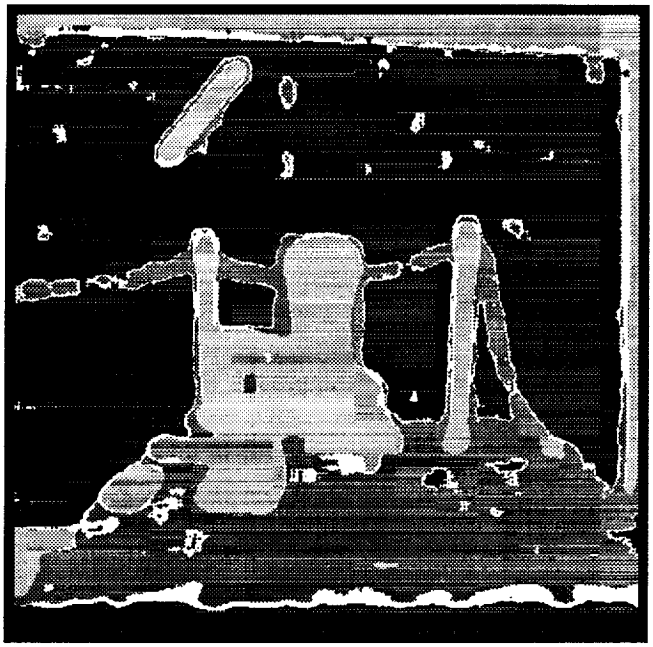

Figure 49. First lab image segmentation with NN using f1, f2, f3, f5, and f13.

The same NN was trained using the f1, f2, f3, f5, and f13 values corresponding to the sample pixels in figure 33. It took 346 cycles to train the NN. The convergence characteristic and the segmentation are shown in figures 51 and 52. In this case 95% of the mountain pixels, 97% of the ground pixels, and 65% of the sky pixels were classified correctly. This is not as good as the correct classification of 98.1% of the mountain pixels, 89.1% of the ground pixels, and 89.4% of the sky pixels, achieved by using f4, f9, f10, f11, and f12 together.

In summary, NNs using the five scalar features f4, f9, f10, f11, and f12 together and f1, f2, f3, f5, and f13 together were trainable for successful classification of the pixels of the images in figures 1 and 33. In both the cases, NNs using the five scalar features together did a better classification when compared to the NNs using a single scalar feature.

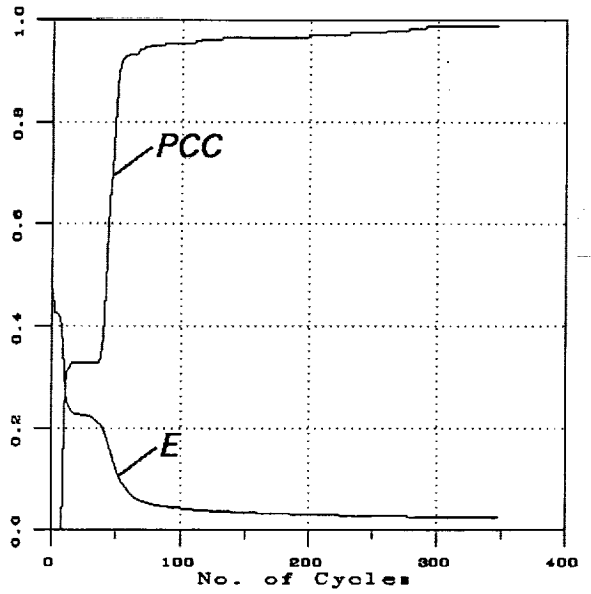

Figure 51. Convergence of NN using f1, f2, f3, f5, and f13 for the UMASS image.

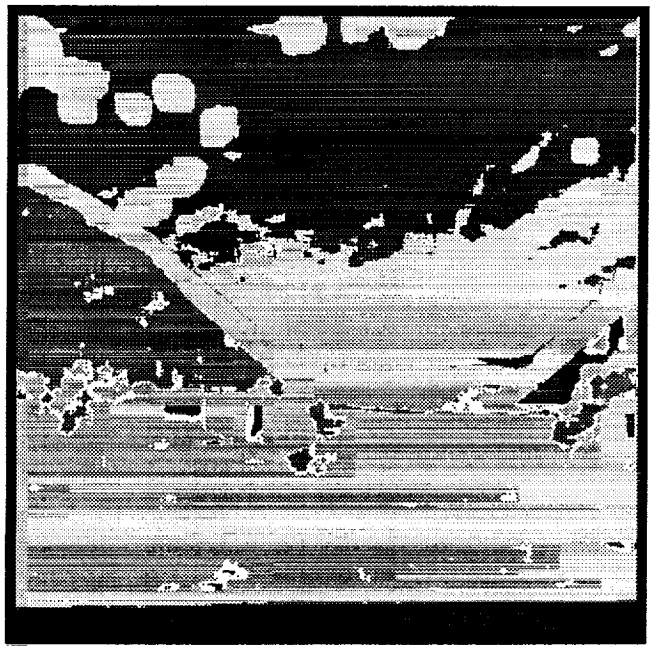

Figure 52. UMASS image segmentation with NN using f1, f2, f3, f5, and f13.

5 CONCLUSIONS

Image segmentations using thresholds derived from histograms of the ten scalar features were described for a laboratory image and an outdoor scene. These scalar features are derived from the spatial gray-level dependence matrix. It was shown that five of these scalar features, namely: variance, entropy, difference variance, difference entropy, and difference average, are individually good descriptors of texture. A neural net was then trained using back propagation with a single scalar feature as input. The performance of the neural network on the training samples and the convergence characteristics were discussed. The trained network was then used for classification of pixels of the whole image. The resulting segmentation result was shown and the neural net classification performance was evaluated. The same neural net was used for classification of pixels of another image in the sequence. By this example, the ability of the trained neural net, using scalar texture features, to generalize to images in the sequence was shown. This further verified that the five scalar features, listed above, are useful for texture segmentation. A neural net was later trained with the five scalar features together. Its convergence characteristics were shown. The trained network was then used for classification of pixels of the whole image. It was shown that the classification results improved considerably using five features together as opposed to using each feature independently. The same network was trained on the lab and outdoor images using the five scalar features, energy, contrast, correlation, and inverse difference moment and mean, which were not found to be useful individual descriptors of texture. In this case also the convergence characteristics were shown for both the images. Generalization to another image in the sequence was also examined using these features. It was shown that these features together are able to correctly classify the image into desired regions. The neural network approach to segmentation using several texture features shows great promise. In the future, we will consider methods to adapt the neural network to improve the generalization. In addition, we will consider alternate neural network schemes.

APPENDIX

The texture features used in this paper for classification are based on the spatial gray-level dependence matrix (SGLDM) (ref. 22). The use of SGLDM to compute texture features involves large memory and computation requirements. For example, 256×256 locations are needed to store SGLDM for an image containing 256 gray-levels. The large dimensionality of the SGLDM makes them sensitive to the sample size from which they are estimated. An alternate to the use of SGLDM is to approximate them by sum and difference histograms. This is based on the observation that a joint probability function of two variables can be approximated by the product of two density functions of uncorrelated transformed variables (ref. 30). This enables us to compute most of the texture features described in reference 22 by using the sum and difference histograms.

Let $p(i,j)$, $p_s(k)$ and $p_d(l)$ be the SGLDM, sum histogram, and the difference histogram, respectively. The sum and difference histograms are generated by computing the sum of gray-levels and the difference of the gray-levels of every horizontal and vertical pixel pair in the region of interest. The indices (i,j) for the SGLDM vary from 0–255 while the indices k, l for the sum and difference histogram vary from 0–512 and 0–255 respectively. Definitions of the 13 texture features are given below.

1. Energy:

$$f1 = \sum_i \sum_j [p(i,j)]^2$$
$$\approx \sum_k [p_s(k)]^2 \sum_l [p_d(l)]^2$$

2. Contrast:

$$f2 = \sum_n n^2 \sum_i \sum_j p(i,j)$$
$$= \sum_l l^2 p_d(l)]$$

Here, $|i-j| = n$ where, n varies from 0-255. This is a weighted sum of the diagonals of the SGLDM where, n^2 is the weight.

3. Correlation:

$$f3 = \frac{\sum_i \sum_j (ij) p(i,j) - f13^2}{f4}$$
$$= \frac{1}{4}[\sum_k (k - 2f13)^2 p_s(k) - \sum_l l^2 p_d(l)]$$
$$= \frac{f7 - f2}{f7 + f2}$$

4. Variance:

$$f4 = \sum_i \sum_j (i - f13)^2 p(i,j)$$
$$= \frac{1}{4}[\sum_k (k - 2f13)^2 p_s(k) + \sum_l l^2 p_d(l)]$$
$$= \frac{1}{4}(f7 + f2)$$

5. Inverse Difference Moment:

$$f5 = \sum_i \sum_j \frac{1}{1 + (i-j)^2} p(i,j)$$
$$= \sum_l \frac{1}{1 + l^2} p_d(l)$$

6. Sum Average:

$$f6 = \sum_k k p_s(k)$$
$$= 2f13$$

7. Sum Variance:

$$f7 = \sum_k (k - f6)^2 p_s(k)$$

8. Sum Entropy:

$$f8 = -\sum_k p_s(k) \log[p_s(k)]$$

9. Entropy:

$$f9 = -\sum_i \sum_j p(i,j) \log[p(i,j)]$$
$$\approx -\sum_k p_s(k) \log[p_s(k)] - \sum_l p_d(l) \log[p_d(l)]$$
$$= f8 + f11$$

10. Difference Variance:

$$f10 = \sum_l (l - f12)^2 p_d(l)$$

11. Difference Entropy:

$$f11 = -\sum_l p_d(l) \log[p_d(l)]$$

12. Difference Average:

$$f12 = \sum_l l p_d(l)$$

13. Mean:

$$f13 = \frac{1}{2} \sum_k k p_s(k)$$
$$= \frac{1}{2} f6$$

REFERENCES

1. Skolnik, M. I.: (ed.), Radar Handbook, McGraw-Hill Pub. Co., San Francisco, 1990.

2. Besl, P. J.; and Jain, R. C.: Segmentation Through Variable-Order Surface Fitting. IEEE Trans. Pattern Anal. Machine Intell., vol. 10, no. 2, Mar. 1988, pp. 167-190.

3. Sridhar, B.; Suorsa, R.; and Hussien, B.: Passive Range Estimation for Rotorcraft Low Altitude Flight. NASA TM-103897, Oct. 1991; also to appear in International J. Machine Vision and Applications, vol. 6, no. 1, 1993.

4. Menon, P. K. A.; Chatterji, G. B.; and Sridhar, B.: Passive Obstacle Location for Rotorcraft Guidance. AIAA Guidance, Navigation, and Control Conference, New Orleans, LA, August 12-14, 1991.

5. Menon, P. K. A.; Chatterji, G. B.; and Sridhar, B.: A Fast Algorithm for Image-Based Ranging. Proc. of SPIE International Symposium on Optical Engineering and Photonics in Aerospace Sensing, April 1-5, 1991, Orlando, FL.

6. Bhanu, B.; Roberts, B.; and Ming, J. C.: Inertial Navigation Sensor Integrated Motion Analysis. Proc. DARPA IUW, May 1989.

7. Chellappa, R.; and Shekar, C.: Passive Ranging using a Moving Camera. J. Robotic Systems, Oct. 1992.

8. Matthies, L.: Stereo Vision for Planetery Rovers: Stochastic Modeling to Near Real-Time Implementation. Jet Propulsion Lab. Rept. No. JPL D-8131, Jan. 1991.

9. Skifstadt, K.; and Jain, R.: Range Estimation from Intensity Gradient Analysis. Machine Vision and Applications, vol. 2, no. 1, 1989, pp. 81-102.

10. Kweon, I. S.; and Kanade, T.: High-Resolution Terrain Map from Multiple Sensor Data. IEEE Trans. Pattern Anal. Machine Intell., vol. 14, no. 2, Feb. 1992, pp. 278-293.

11. Krishnapuram, R.; and Casasent, D.: Determination of Three-dimensional Object Location and Orientation from Range Images. IEEE Trans. Pattern Anal. Machine Intell., vol. 11, no. 11, Nov. 1989, pp. 1158-1167.

12. Sinha, S. S.; and Schunck, B. G.: A Two-Stage Algorithm for Discontinuity-Preserving Surface Reconstruction. IEEE Trans. Pattern Anal. Machine Intell., vol. 14, no. 1, Jan. 1992, pp. 36-55.

13. Sridhar, B.; and Chatterji, G. B.: Object Segmentation for Helicopter Guidance. Proc. of SPIE Conference on Intelligent Information Systems Applications of AI X: Machine Vision and Robotics, April 22-24, 1992, Orlando, FL.

14. Wechsler, H.; and Reed, T. R.: Segmentation of Textured Images and Gestalt Organization Using Spatial/Spatial-Frequency Representations. IEEE Trans. Pattern Anal. Machine Intell., vol. 12, no. 1, Jan. 1990, pp. 1-12.

15. Tüceryan, M.; and Jain, A. K.: Using Voronoi Polygons. IEEE Trans. Pattern Anal. Machine Intell., vol. 12, no. 2, Feb. 1990, pp. 211-216.

16. Pavlidis, T.; and Liow Y.: Integrating Region Growing and Edge Detection. IEEE Trans. Pattern Anal. Machine Intell., vol. 12, no. 3, Mar. 1990, pp. 225-233.

17. Geman, D.; Geman, S.; Graffigne, C.; and Dong, P.: Boundary Detection by Constrained Optimization. IEEE Trans. Pattern Anal. Machine Intell., vol. 12, no. 7, July 1990, pp. 609-628.

18. Berry, J. R.; and Goutsias, J.: A Comparitive Study of Matrix Measures for Maximum Likelihood Texture Classification. IEEE Transactions on Systems, Man, and Cybernetics, vol. 21, no. 1, Jan-Feb, 1991.

19. Ahuja, N.; and Rosenfeld, A.: A Note on the Use of Second-Order Gray-Level Statistics for Threshold Selection. IEEE Transactions on Systems, Man, and Cybernetics, vol. 8, no. 12, Dec, 1978.

20. Chen, P. C.; and Pavlidis, T.: Segmentation by Texture Using A Co-Occurrence Matrix and a

Split-and-Merge Algorithm. Computer Graphics and Image Processing, vol. 10, 1979, pp. 172-182.

21. Xu, G. Y.; and Fu, K. S.: Natural Scene Segmentation Based on Multiple Threshold and Textural Measurement. IEEE Conference,1984.

22. Haralick, R. M.; Shanmugam, R.; and Dienstein, I.: Textural features for Image Classification. IEEE Transactions on Systems, Man, and Cybernetics, vol. 3, no. 6, November, 1973.

23. Vickers, A. L.; and Modestino, J. W.; A Maximum Likelihood Approach to Texture Classification. IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI, vol. 4, no. 1, Jan, 1982.

24. Haddon, J. F.; and Boyce, J. F.: Image Segmentation by Unifying Region and Boundary Information. IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI, vol. 12, no. 10, Oct, 1990.

25. Rosenfeld, A.; and Davis, L. S.: Image Segmentation and Image Models. Proc. of the IEEE, vol. 67, no. 5, May, 1979, pp. 764-772.

26. Harlick, R. M.: Statistical and Structural Approach to Texture. Proc. of the IEEE, vol. 67, no. 5, May, 1979, pp. 764-772.

27. Rounds, E. M.; and Sutty, G.: Segmentation based on second-order statistics. Proc. of the SPIE, Image Understanding Systems II, vol. 205, 1979, pp. 126-132.

28. Conners, R. W.; and Harlow, C. A.: A Theoretical Comparison of Texture Algorithms. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 2, no. 3, May, 1980, pp. 204-222.

29. Weszka, J. S; Dyer, C. R.; and Rosenfeld, A.: A Comparitive Study of Texture Measures for Terrain Classification. IEEE Transactions on Systems, Man, and Cybernetics, vol. 6, no. 4, Apr. 1976, pp. 269-285.

30. Unser, M.: Sum and Difference Histograms for Texture Classification. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 8, no. 1, Jan., 1986, pp. 118-125.

31. Dutta, R.; Manmatha, R.; Williams, L. R.; and Riseman, E. M.: A Data Set for Quantitative Motion Analysis. Proceedings of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, San Diego, CA, pp. 159-164, June 4-8, 1989. no. 1, Jan., 1986, pp. 118-125.

32. Rumelhart, D. E.; and McClelland, J. L.: (eds.), Parallel Distributed Processing: Explorations in the Microstructure of Cognition, vol. 1: Foundations, MIT Press, 1986.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	March 1993	Technical Memorandum

4. TITLE AND SUBTITLE

Scene Segmentation of Natural Images Using Texture Measures and Back-Propagation

5. FUNDING NUMBERS

505-64-36

6. AUTHOR(S)

Banavar Sridhar, Anil Phatak, and Gano Chatterji

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Ames Research Center
Moffett Field, CA 94035-1000

8. PERFORMING ORGANIZATION REPORT NUMBER

A-93043

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

National Aeronautics and Space Administration
Washington, DC 20546-0001

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

NASA TM-104003

11. SUPPLEMENTARY NOTES

Point of Contact: Banavar Sridhar, Ames Research Center, MS 210-9, Moffett Field, CA 94035-1000
(415) 604-5440

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Unclassified-Unlimited
Subject Category – 59

12b. DISTRIBUTION CODE

13. ABSTRACT ***(Maximum 200 words)***

Knowledge of the three-dimensional world is essential for many guidance and navigation applications. A sequence of images from an electro-optical sensor can be processed using optical flow algorithms to provide a sparse set of ranges as a function of azimuth and elevation. A natural way to enhance the range map is by interpolation. However, this should be undertaken with care since interpolation assumes continuity of range. The range is continuous in certain parts of the image and can jump at object boundaries. In such situations, the ability to detect homogeneous object regions by scene segmentation can be used to determine regions in the range map that can be enhanced by interpolation. This paper explores the use of scalar features derived from the spatial gray-level dependence matrix for texture segmentation. Thresholding of histograms of scalar texture features is done for several images to select scalar features which result in a meaningful segmentation of the images. Next, the selected scalar features are used with a neural net to automate the segmentation procedure. Back-propagation is used to train the feed forward neural network. The generalization of the network approach to subsequent images in the sequence is examined. It is shown that the use of multiple scalar features as input to the neural network result in a superior segmentation when compared with a single scalar feature. It is also shown that the scalar features, which are not useful individually, result in a good segmentation when used together. The methodology is applied to both indoor and outdoor images.

14. SUBJECT TERMS

Scene segmentation, Neural networks, Helicopter guidance

15. NUMBER OF PAGES

25

16. PRICE CODE

A02

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified		